

Contents

1	Classes	3
1.1	real – real numbers and its functions	3
1.1.1	RealField – field of real numbers	5
1.1.1.1	getCharacteristic – get characteristic	6
1.1.1.2	issubring – subring test	6
1.1.1.3	issuperring – superring test	6
1.1.2	Real – a Real number	7
1.1.2.1	getRing – get ring object	8
1.1.3	Constant – real number with error correction	9
1.1.4	ExponentialPowerSeries – exponential power series	9
1.1.5	AbsoluteError – absolute error	9
1.1.6	RelativeError – relative error	9
1.1.7	exp(function) – exponential value	9
1.1.8	sqrt(function) – square root	9
1.1.9	log(function) – logarithm	9
1.1.10	log1piter(function) – iterator of log(1+x)	9
1.1.11	piGaussLegendre(function) – pi by Gauss-Legendre	9
1.1.12	eContinuedFraction(function) – Napier’s Constant by continued fraction expansion	9
1.1.13	floor(function) – floor the number	10
1.1.14	ceil(function) – ceil the number	10
1.1.15	trunc(function) – round-off the number	10
1.1.16	sin(function) – sine function	10
1.1.17	cos(function) – cosine function	10
1.1.18	tan(function) – tangent function	10
1.1.19	sinh(function) – hyperbolic sine function	10
1.1.20	cosh(function) – hyperbolic cosine function	10
1.1.21	tanh(function) – hyperbolic tangent function	10
1.1.22	asin(function) – arc sine function	11
1.1.23	acos(function) – arc cosine function	11
1.1.24	atan(function) – arc tangent function	11
1.1.25	atan2(function) – arc tangent function	11
1.1.26	hypot(function) – Euclidean distance function	11
1.1.27	pow(function) – power function	11

1.1.28	degrees(function) – convert angle to degree	11
1.1.29	radians(function) – convert angle to radian	11
1.1.30	fabs(function) – absolute value	11
1.1.31	fmod(function) – modulo function over real	11
1.1.32	frexp(function) – expression with base and binary exponent	12
1.1.33	ldexp(function) – construct number from base and binary exponent	12
1.1.34	EulerTransform(function) – iterator yields terms of Euler transform	12

Chapter 1

Classes

1.1 `real` – real numbers and its functions

The module `real` provides arbitrary precision real numbers and their utilities. The functions provided are corresponding to the `math` standard module.

- **Classes**

- `RealField`
- `Real`
- †`Constant`
- †`ExponentialPowerSeries`
- †`AbsoluteError`
- †`RelativeError`

- **Functions**

- `exp`
- `sqrt`
- `log`
- `log1piter`
- `piGaussLegendre`
- `eContinuedFraction`
- `floor`
- `ceil`
- `trunc`
- `sin`
- `cos`

- **tan**
- **sinh**
- **cosh**
- **tanh**
- **asin**
- **acos**
- **atan**
- **atan2**
- **hypot**
- **pow**
- **degrees**
- **radians**
- **fabs**
- **fmod**
- **frexp**
- **ldexp**
- **EulerTransform**

This module also provides following constants:

- e** :
This constant is obsolete (Ver 1.1.0).
- pi** :
This constant is obsolete (Ver 1.1.0).
- Log2** :
This constant is obsolete (Ver 1.1.0).
- theRealField** :
theRealField is the instance of **RealField**.

1.1.1 RealField – field of real numbers

The class is for the field of real numbers. The class has the single instance **theRealField**.

This class is a subclass of **Field**.

Initialize (Constructor)

RealField() → *RealField*

Create an instance of RealField. You may not want to create an instance, since there is already **theRealField**.

Attributes

zero :

It expresses the additive unit 0. (read only)

one :

It expresses the multiplicative unit 1. (read only)

Operations

operator	explanation
x in R	membership test; return whether an element is in or not.
repr(R)	return representation string.
str(R)	return string.

Methods

1.1.1.1 `getCharacteristic` – get characteristic

`getCharacteristic(self)` → *integer*

Return the characteristic, zero.

1.1.1.2 `issubring` – subring test

`issubring(self, aRing: Ring)` → *bool*

Report whether another ring contains the real field as subring.

1.1.1.3 `issuperring` – superring test

`issuperring(self, aRing: Ring)` → *bool*

Report whether the real field contains another ring as subring.

1.1.2 Real – a Real number

Real is a class of real number. This class is only for consistency for other **Ring** object.

This class is a subclass of **CommutativeRingElement**.

All implemented operators in this class are delegated to Float type.

Initialize (Constructor)

Real(value: *number*) → *Real*

Construct a Real object.

value must be int, long, Float or **Rational**.

Methods

1.1.2.1 `getRing` – get ring object

`getRing(self)` → *RealField*

Return the real field instance.

1.1.3 Constant – real number with error correction

This class is obsolete (Ver 1.1.0).

1.1.4 ExponentialPowerSeries – exponential power series

This class is obsolete (Ver 1.1.0).

1.1.5 AbsoluteError – absolute error

This class is obsolete (Ver 1.1.0).

1.1.6 RelativeError – relative error

This class is obsolete (Ver 1.1.0).

1.1.7 exp(function) – exponential value

This function is obsolete (Ver 1.1.0).

1.1.8 sqrt(function) – square root

This function is obsolete (Ver 1.1.0).

1.1.9 log(function) – logarithm

This function is obsolete (Ver 1.1.0).

1.1.10 log1piter(function) – iterator of $\log(1+x)$

log1piter(xx: *number*) → *iterator*

Return iterator for $\log(1+x)$.

1.1.11 piGaussLegendre(function) – pi by Gauss-Legendre

This function is obsolete (Ver 1.1.0).

1.1.12 eContinuedFraction(function) – Napier's Constant by continued fraction expansion

This function is obsolete (Ver 1.1.0).

1.1.13 floor(function) – floor the number

floor(*x: number*) → *integer*

Return the biggest integer not more than *x*.

1.1.14 ceil(function) – ceil the number

ceil(*x: number*) → *integer*

Return the smallest integer not less than *x*.

1.1.15 trunc(function) – round-off the number

trunc(*x: number*) → *integer*

Return the number of rounded off *x*.

1.1.16 sin(function) – sine function

This function is obsolete (Ver 1.1.0).

1.1.17 cos(function) – cosine function

This function is obsolete (Ver 1.1.0).

1.1.18 tan(function) – tangent function

This function is obsolete (Ver 1.1.0).

1.1.19 sinh(function) – hyperbolic sine function

This function is obsolete (Ver 1.1.0).

1.1.20 cosh(function) – hyperbolic cosine function

This function is obsolete (Ver 1.1.0).

1.1.21 tanh(function) – hyperbolic tangent function

This function is obsolete (Ver 1.1.0).

1.1.22 `asin(function)` – arc sine function

This function is obsolete (Ver 1.1.0).

1.1.23 `acos(function)` – arc cosine function

This function is obsolete (Ver 1.1.0).

1.1.24 `atan(function)` – arc tangent function

This function is obsolete (Ver 1.1.0).

1.1.25 `atan2(function)` – arc tangent function

This function is obsolete (Ver 1.1.0).

1.1.26 `hypot(function)` – Euclidean distance function

This function is obsolete (Ver 1.1.0).

1.1.27 `pow(function)` – power function

This function is obsolete (Ver 1.1.0).

1.1.28 `degrees(function)` – convert angle to degree

This function is obsolete (Ver 1.1.0).

1.1.29 `radians(function)` – convert angle to radian

This function is obsolete (Ver 1.1.0).

1.1.30 `fabs(function)` – absolute value

`fabs(x: number) → number`

Return absolute value of `x`

1.1.31 `fmod(function)` – modulo function over real

`fmod(x: number, y: number) → number`

Return $x - ny$, where `n` is the quotient of `x / y`, rounded towards zero to an integer.

1.1.32 frexp(function) – expression with base and binary exponent

frexp(x: number) → (m,e)

Return a tuple (m,e), where $x = m \times 2^e$, $1/2 \leq \text{abs}(m) < 1$ and e is an integer.

†This function is provided as the counter-part of math.frexp, but it might not be useful.

1.1.33 ldexp(function) – construct number from base and binary exponent

ldexp(x: number, i: number) → number

Return $x \times 2^i$.

1.1.34 EulerTransform(function) – iterator yields terms of Euler transform

EulerTransform(iterator: iterator) → iterator

Return an iterator which yields terms of Euler transform of the given iterator.

Bibliography